(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
15 March 2001 (15.03.2001)

PCT

(10) International Publication Number
WO 01/18619 A1

(51) International Patent Classification[7]: G05B 19/042

(21) International Application Number: PCT/FI00/00692

(22) International Filing Date: 15 August 2000 (15.08.2000)

(25) Filing Language: Finnish

(26) Publication Language: English

(30) Priority Data:
19991890 3 September 1999 (03.09.1999) FI

(71) Applicant (*for all designated States except US*): EUROELEKTRO INTERNATIONAL OY [FI/FI]; Puusepäntie 5, FIN-54100 Joutseno (FI).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): PULKKINEN, Veijo [FI/FI]; Puusepäntie 5, FIN-54100 Joutseno (FI).

(74) Agent: PATENT AGENCY COMPATENT LTD.; Pitkänsillanranta 3 B, FIN-00530 Helsinki (FI).

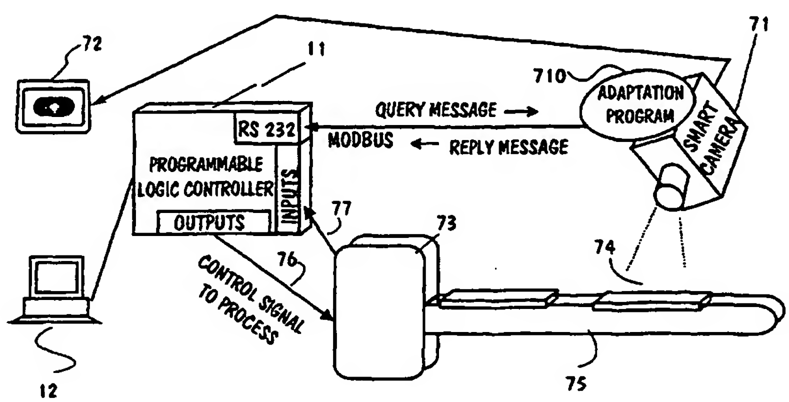
(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: CAMERA CONTROL IN A PROCESS CONTROL SYSTEM



(57) Abstract: When modifications in a process control system are made, the software modifications must be made in the camera linked to the system, too. Both a programmer skilled in the process control system and a programmer skilled in camera programming are needed. This can be avoided when an adaptation program (710) is made for a smart camera (71). This program is able to transform the tasks given by the programmable logic (11) to a language understood by the camera software as well as to send the task results to the programmable logic. The command tasks are transmitted from the programmable logic to the smart camera, and correspondingly, the test results are transmitted to the programmable logic in messages of a known field bus protocol (e.g. Modbus), wherein the adaptation program acts as an interpreter between the bus protocol used and the specific camera software. In addition to the adaptation program another program is made for the programmable logic; this program may include any tasks to be given to the camera image-processing program provided that the tasks are incorporated in the adaptation program. The logic (11) program can now be modified at any time on the condition mentioned above, it can be included new tasks or the parameters of the existing tasks can be modified without any need for modifications in the camera (71) software or in the adaptation program.

WO 01/18619 A1

CAMERA CONTROL IN A PROCESS CONTROL SYSTEM

FIED OF THE INVENTION

The present invention relates to a process control system and a camera monitoring a process. A video picture produced by the camera is first processed and then the process results are transmitted via a data transfer system to the control system.

BACKGROUND OF THE INVENTION

The programmable logic controllers (PLC) have developed from the simple devices in the early 1970's that used integrated circuit technology and were able to carry out simple repeating control tasks to small and complex systems able to perform nearly all kinds of control applications requiring ability to data processing and advanced computation. They can be integrated to large systems in which various logic units communicate with each other as well as with computers controlling operation of a factory. Industrial applications for PLCs can be found especially in production, petrochemistry, construction industry, and food and beverage industries, where they control temperatures and electro-mechanical devices, such as valves, conveyors etc.

A programmable logic used as a control unit may include dozens or hundreds of I/O ports. Typical devices connected to input ports are press buttons, limit switches, proximity switches, and temperature sensors. Solenoids, motors, contactors etc. can be connected to output ports. In brief, the PCL operates by scanning its inputs and registering their states. After that it carries out its program and changes, following the instructions of the program, the states of the outputs to ON or OFF state. Then the inputs are scanned again and it is checked whether there are changes in the states or not. Finally the program is carried out again and the states of the inputs are changed in the way the program commands. The entire scanning cycle takes usually 1 to 40 ms, but depends naturally on the length of the program and on how long time it takes to carry out the commands. Once the program has been written, it 's easy to bring it into use: the needed devices are connected

to the input and output ports, whereupon a complete process control system has been created.

In many applications, especially in those monitoring the shape, dimensions and location of a product, machine vision is of great advantage. A CCD camera (Charge Coupled Device) is mainly used, on the sensors of which a picture of the target is formed. The analog signal is converted into a digital one and transmitted to an image-processing card where different image-processing operations take place.

When a camera picture is connected to the input of a programmable logic controller, the process control system illustrated in FIG. **1** is obtained. The processes to be controlled consist of various functions that need to be controlled and adjusted. One of these functions has been marked with broken line **14** in FIG. **1**. Actuator **15** that may be e.g. a burner of an oven regulates the process. A sensor measures a value for the regulating unit, in this case the temperature of the oven. The signal of the sensor is led to programmable logic **11**. After the scanning period the logic gives the control to actuator **15** if the program tells to do so.

CCD camera **13** is used as a sensor and it has been connected to computer **16** via a video bus. The computer contains an image-processing card and it may be a common multipurpose PC. The computer processes the image according to the program and gives the results to programmable logic **11** via a connection bus. In this example, the CCD camera would monitor the shape and color of the products coming out of the oven.

A system similar to FIG. **1** has been presented in the patent application DE-4325325. There the logic is programmed with a programming device consisting of a keyboard, a monitor and a CPU. The video input of the programming device has been cabled to the video output of a remote process control camera. The video picture from the camera can be seen on the display and the logic can then be programmed as desired. In this case the video camera serves only as the user's visual aid.

A so-called smart camera can be used as well. It contains circuits and software needed for image-processing. An image can then be processed in the camera itself, and no computer is needed. Programming of a smart camera depends on the manufacturer, which means that a programmer specialized in each type of camera is needed. In many cases the cam-

era is supplied to the user ready programmed according to his special wishes.

Programmable logics are able to intercommunicate and exchange messages via a field bus. The most common field bus protocols are Modbus and Profibus, the latter having been specified in standard EN 50170. The protocols define the message structure very precisely, and the devices are classified to master and slave devices. Modbus uses RS 232C and Profibus mainly RS 485 transfer technology.

In order to give a clarifying example the Modbus protocol will be explained here more in detail. The protocol defines how a device knows its own device address, recognizes a message addressed to it, knows what functions it has to do and is able to distinguish data from a message. If the data transfer system is other than the Modbus, e.g. the Ethernet or the TCP/IP network, the messages are embedded into the frames or packets of the network in question. Communication always takes place using the master-slave principle, i.e. only the master device is able to start the transactions whereas the slave device responds by sending the requested data or by performing the functions asked by the master. Usually the master is a programmable logic and the slave device a peripheral, such as an I/O device, a valve, a driver or a measuring apparatus.

In FIG. **2** a message structure in accordance with the Modbus protocol is presented. The message starts with a "Start" sign, i.e. a semicolon, and ends to an "End" sign which is a CRLF (Carriage Return-Line) sign. After the start sign there is the individual address of the target device. When replying to the message the target device relocates its own address in this field, on the basis of which the master device knows where the reply had come from.

The code to be given in the "function" field, which comes next, may have the values from 1 to 255 (given in decimal numbers). It tells what kind of a function the slave device has to perform. When giving an answer the target device uses this field in order to indicate either that the answer was correct or that an error had occurred during the performance of the task.

In the next field, the"data" field, the master device gives information needed by the slave device for performing the given task. When replying the slave device relocates the data resulted from the performed task in this field.

The error checking field contains the error checking of the message contents, either a LRC or a CRC error check sum.

FIG. **3** presents a case where a camera as a peripheral is connected to the programmable logic using a known field bus. The logic and computer **16** have RS 232 connections. Unlike in FIG. **1** there is a bidirectional connection between the computer and the programmable logic, in this case the Modbus. This means that the logic can ask the computer when it has finished its computations. The program carries out in advance the image-processing programmed into it, and the logic cannot interfere in any way. It only receives the results.

A system like this is presented in the patent US-5882402. The process consists of drawing optical fiber out of a molten crystal. The image-processing unit has an image processor and it is attached to a camera that monitors the fiber. It is connected to the programmable logic by means of a duplex data transfer channel. The logic is programmed via the operator's computer. The image-processing, i.e. the camera, is programmed via a computer connected to the image-processing unit. One possible procedure is to merge the computer connected to the image-processing unit of the camera with that of the operator. According to the specification the merged computer acts as a user interface in order to program the programmable logic. Here we can, at least in theory, program both the logic and the image-processing of the camera using the same user interface. However, the programs of each device are to that extent different that the process operator must have a deep knowledge of the camera software.

FIG **4** presents a case in which the camera and the computer have been replaced with so-called smart camera **41**. The field bus is directly connected to the interface of the camera. In this case, too, the camera program processes the image in advance according to the program and the logic cannot interfere in any way. It only receives the results.

The problem with these two cases, where the camera is connected to the programmable logic via a field bus, is that the smart camera or the computer must be made able to perform the tasks the logic tells them to do and to give the task results the logic tells them to give. The camera is programmed to give only the requested results, and if any other information is required, the camera has to be reprogrammed. The programs for the computer and the smart camera differ from one software vendor to another,

which means that they require special know-how. The manufacturer very often programs the device according to the customer's instructions prior to supply. If the customer wishes to have any modifications in the process to be controlled or if he wishes that the camera gives other values than before, changes in the camera software must be made. Then the supplier is requested to send a specialist programmer to do the modifications. On the other hand, the programming of a PLC requires expertise and know-how that the process supervisor must naturally have. When the process supervisor wants that the camera gives new information as a response to a new command, prolonged cooperation is often needed between the process supervisor, who knows to PCL program, and the programmer specialized in programming cameras. This is both expensive and time-consuming, as the reprogramming costs are high and two persons are needed to do the work: one takes care of the camera, the other of the computer. It may be difficult to come to mutual understanding.

There is an attempt to overcome this problem by making it easier to (re)program a smart camera, namely by using different types of graphical interfaces. There are programs of Windows® type suitable for various machine vision applications. These programs, e.g. AEInspect and FlexAuto for Windows, made by Automation Engineering Inc. USA, work in common multipurpose PCs. Despite these programs are apparently easy to use, they as well as programming with the use of them require a high level of expertise that a process supervisor hardly has. Additionally, one must literally go to the camera, be it at a great distance or not, if the program has to be modified.

SUMMARY OF THE INVENTION

An objective of the present invention is to devise a camera control connected to a process control unit, especially to the programmable logic controller, so that the control does not have the drawbacks of the prior art systems. The objective is a system in which the operator of the control unit can easily program the computer connected to the camera or the smart camera so that the computer of the camera can perform tasks defined in a query message sent via a data transfer channel as well as relocate the task results in the reply message.

Firstly, the invention is based on the idea that the computer linked with a smart camera or an ordinary camera can be provided with a adaptation program. This program is able to compile the tasks given by the process control unit to a language understood by the camera software. Correspondingly, it can send the task results to the control unit. The amount of parameters needed for the tasks and their performance is only limited by the ability of the camera software to carry out the tasks.

Another idea is to transmit the command tasks from the process control unit to the smart camera or the computer and the task results Hence, the adaptation program acts as an compiler between the protocol used and the specific software of the camera. As a data transfer connection it is favorable to choose a modern field bus using the Modbus or the Profibus protocol.

Before the user of the programmable logic brings the system into use he makes a covering list of all image-processing tasks he wants the camera software to perform. Each task gets a code, which is a number, for example. Additionally, each program is provided with an adequate number of parameters needed for task performance. After that a adaptation program is made. It is able to distinguish each task code and parameter from the message that is in accordance with the used data transfer protocol and to compile these into a form understood by the specific camera software. The adaptation program gives the task code with its parameters to the camera software. Then the software performs the task according to the task code and returns the task results to the adaptation program in a form that it is able to understand. After this the adaptation program forms a reply message according to the used data transfer protocol, relocates the results in the reply message and transmits the message to the transmission channel. The only limitation for the tasks given by the adaptation program to the image-processing program is the capacity of the latter to do the tasks given to it.

Besides the adaptation program another program is made in the process control unit, e.g. in the programmable logic. In this program any tasks for to the image-processing program can be included provided that the tasks are incorporated in the adaptation program. The program of the control unit can now be modified according to the condition mentioned above whenever there is a need. New tasks can be included in it or the parameters of existing tasks can be changed without any need to modify the camera software or the adaptation program.

If desired, the image signal from the camera can be led, using a separate connection, to a monitor in the process supervisor's facilities. Then the supervisor sees the picture of the target and is able to give various tasks to the camera in a very flexible manner. It is easiest to give the tasks via the same user interface with which the process control unit is normally programmed.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is described more in detail with reference to the accompanied schematic drawings in which

Fig. 1 shows a known process control system in which the camera is used as a sensor;
Fig. 2 shows message fields according to the Modbus protocol;
Fig. 3 shows a known system in which a camera linked to the computer is connected to a programmable logic via a field bus;
Fig. 4 shows a known system using a smart camera;
Fig. 5 illustrates in broad outline how a adaptation program is developed;
Fig. 6 shows schematically functions of a proceeding logic program;
Fig. 7 illustrates an embodiment of the system based on the invention;
Fig. 8 shows a picture to be examined;
Fig. 9 is a structure of the query message;
Fig. 10 is a structure of the reply message;
Fig. 11 shows another embodiment of the based on the invention, and
Fig. 12 is a partial enlargement of Fig. 11

DETAILED DESCRIPTION OF THE INVENTION

Fig. **5** illustrates how a system based on the invention is brought into use. When a camera is connected to a programmable logic or a factory system one has to define what kind of information the image-processing software of the camera should produce, step **51**. The desired information depends naturally on the target of the camera. If the target is for instance an object with an additional part, that area of the picture is viewed where the additional part is supposed to be. In this area the image-processing program

examines whether the part has been attached or if it is missing. Hence, the number and nature of the tasks to be performed depends on each project.

When the tasks have been determined, an individual code, a task number and the needed parameters are attached to each task, step **52**. For instance, task number 1 could signify that the average shade of gray of a picture should be calculated. The parameter associated with this task would give the pixel density used in calculation; the parameter value would e.g. mean that every fourth pixel ought to be counted.

When all the tasks have been determined, equipped with an code and parameters, an applications program can be written. This program understands what the image-processing program has to do, step **53**, on the basis of the figures in the data field that have been sent in a message according to the bus protocol. The adaptation program must know how to compile the tasks defined above to a language understood by the camera software in such a manner that the software is able to carry out the tasks. The adaptation program also has to be able to receive the results from the camera program and to relocate them correctly in a reply message according to the data transfer protocol, as well as to pass the message to a data transfer channel.

When the adaptation program has been accomplished it is installed to the camera, step **54**. After the program has been programmed in the process control unit that uses the above mentioned task numbers and parameters in the messages that it sends to the camera, the system is ready for use. The process operator now easily determines what the camera software does by changing the values of the parameters at any time.

From now on we use as an example of the process control unit the programmable logic and as an example of a data transfer connection the known field bus with the Modbus protocol.

In Fig. **6** the function of the system is illustrated during the process. Let us suppose that the software of the logic has proceeded to a point where it needs information of the camera picture, step **61**. Then the software edits a query according to the bus protocol used, step **62**. It puts in this message the task code and parameters, the realization of which provides the information needed by the logic program. When the query is ready, it is sent to the bus, step **63**.

The adaptation program recognizes the query addressed to it from the device address and opens it, step **64**. The adaptation program knows of

the task code of the message what the camera program has to do. So, it gives the task and the related parameters to the camera program, step **65**. The camera program carries out the given image-processing task, step **66**, and returns the results to the adaptation program, step **67**. It edits a reply according to the bus protocol and relocates the results from the camera program into it, step **68**.

During the previous steps the logic program has regularly polled the devices connected to the bus. When such a polling comes to the camera it sends a reply, step **69**. The logic receives the message from the bus and recognizes the task results in the data part, step 610, and then gives this result to point **61** of the logic program that has asked for this information.

In the logic program there may be several points that need information about the picture taken by the camera. The program may also ask the camera program more precise questions on the basis of received information. The program in the logic utilizes the task reply in a manner required by the program. Depending on application the program may ask the camera one or more additional questions before the process is influenced or any decision made. A conveyor could serve as an example. It may convey objects of five different sizes. The system based of the invention should decide whether the object is acceptable or should it be rejected. When the camera has viewed the object and its picture has been transmitted to the program, the logic sends, e.g. triggered by a photocell, the first query to the camera asking about the dimensions of the object. The camera program calculates the dimensions from the picture and the application program sends the information to the logic. On the basis of the information the logic program concludes which one of the five objects is concerned. After this the logic program branches out to the program branch concerning this very object and may ask the camera many additional questions. As a response to the queries the camera software calculates the required data from the saved picture and sends them in replies to the programmable logic. In this way several picture details can be checked and finally the conclusion can be made whether the object should be accepted or not. If the object is to be rejected the programmable logic gives the related output the signal of removing the object from the belt.

Queries can either be sent periodically using the scanning principle or they are sent only if a certain triggering limit has been exceeded. There

may be many different queries. The logic program decides which message is sent in each case. In different applications the query type to be sent may depend on the reply to the previous query.

When the operator of a process control system wants to make changes in the programmable logic he makes this by means of the programming device, as we know. If the program modifications to be made require that the image-processing program of the camera performs other tasks and gives other results than those defined in the queries used so far, the maintainer sets a new query in the logic program, or better said a task code for a query with the related parameters. If the query code is already familiar to the adaptation program, there is no need for modifying the adaptation program in the camera.

It is essential to notice that the same person who programs the logic also "programs" the camera. There is no need for a specialist in camera programming to make program modifications in order to obtain new type of results.

FIRST EMBODIMENT OF THE INVENTION

Fig. **7** shows a system based on the invention in an environment in which the quality of targets **74** coming from device **73** to conveyor **75** has to be controlled. Device **73** may be an assembly device, a cutter or the like. the function element of which, e.g. a press, positioning element or the like (not shown), is controlled by means of control signal **76** given by programmable logic **11**. To simplify, it is supposed that the device in question is a perforating machine that perforates metal sheets.

Initializing data **77** received from the function element is led to one of the input ports of the logic. The device feeds finished products continuously onto conveyor **75** that brings them forward. In the figure the products are roughly drawn as rectangular pieces. Camera **71** monitors the pieces. It has adaptation program 710 based on the invention and its task is to control that the perforation is made correctly. When a piece **74** is within the shooting area, the sensor e.g. a photocell (not presented in the figure) tells the camera to take a picture. The picture is stored in the memory of the camera. It can immediately be viewed on monitor **72** that is located in the same facilities as the process control. The monitor is connected with a long separate cable to the video interface of the camera.

Fig. **8** is an allusive illustration of the camera picture. The coordinates X and Y have the values from 0 to 100. Let us suppose that four rectangular areas **1**, **2**, **3**, and **4** have to be viewed more in detail. Programmable logic **11** passes, triggered by the sensor signal, a query via the Modbus to the adaptation program and tells the camera software to calculate the data given in the message. The query is about the average of the gray scale values in area **1**. This can be used for checking the camera settings. e.g. the gain. The code of this task is **1**. Referring to area **2** the dimension of hole **81** shall be determined. The code of this task is **2**. Concerning area **3** the exact location of the center shall be determined. The code of the task is **4**. About area **4** one would like to know between which X coordinates slot **83** is situated, i.e. whether the slot is correctly positioned. The code number of the task is **5**.

The structure of the query message has been presented in Fig. **9**. The upper part of the figure shows the Modbus message described above. The lower part presents its data field where the data needed by the camera for task completion is located. The task is to analyze a picture showing the object, and there are four areas in this picture that must be analyzed, Fig. **8**. Therefore the message informs the number of areas to be analyzed, what the areas are, and what exactly must be examined in each area. This information is in the data field in the following order: First there is the number of areas to be examined, i.e. four. Next the size of the areas to be examined is given, by using the X and Y coordinates in succession, first the coordinates of area **1** and finally those of area **4**. Hence, to define one rectangular area two X and two Y coordinates are needed, so the length of one area field is four bytes. Thirdly, task definitions for each area are enumerated in succession.

Table **1** shows, for the sake of clarity, one possible value sequence in the data field.

Value	Description	Position in fig.9
4	Number of the areas to be analyzed	Number of the areas to be analyzed
20	X coordinate of 1st area	Coordinates of the 1st area
30	X coordinate of the1st area	
15	Y coordinate of the 1st area	
30	Y coordinate of the 1st area	
40	X coordinate of the 2nd area	Coordinates of the 2nd area
45	X coordinate of the 2nd area	
5	Y coordinate of the 2nd area	
30	Y coordinate of the 2nd area	
55	X coordinate of the 3rd area	Coordinates of the 3rd area
60	X coordinate of the 3rd area	
45	Y coordinate of the 3rd area	
46	Y coordinate of the 3rd area	
70	X coordinate of the 4th area	Coordinates of the 4th area
90	X coordinate of the 4th area	
20	Y coordinate of the 4th area	
90	Y coordinate of the 4th area	
1	Average gray scale value in the 1st area	Task relating to the 1st area
4	Every 4th pixel in calculation	
2	Surface area in the 2nd area having gray scale value greater than the average	Task relating to the 2nd area
4	Mass center point of the surface area in the 3rd area having gray scale value greater than the average	Task relating to the 3rd area
5	Boundaries of the dark sub-area in the 4th area	Task relating to the 4th area
5	Both x coordinates must be included in the reply message	

Table 1

In the first data position there is the value 4 that refers to the number of rectangular areas to be examined. In the four next data positions X

coordinates (20, 30) and the Y coordinates (15, 30) of the first area to be examined are given.

The value 1 in the data position for the task definition of the first area indicates that the average value of the shade of gray must be calculated. The value 4 of the next data position informs that only every fourth pixel is calculated. It worth noting that the average gray scale value that has been calculated picture by picture (product by product) is moving. Its advantage is that external conditions, such as a change in the lighting or dirt, do not affect the result, as when areas lighter or darker than the average shade of gray are calculated they are compared to the average shade of gray of the same picture.

The task to be carried out in area 2 is to calculate the area having the level of gray greater the average level of gray. This gives the dimensions of hole **81**, fig. **8**. The value 2 of the data position indicates this task.

The task to be carried out in area 3 is to calculate the mass center of the area having the level of gray greater the average level of gray, i.e. the center of hole **82**, fig. **8**. The task can be indicated by using one data position and placing the value 4 in it.

The two last data position values, 5 and 5, indicate the task to be carried out in area 4. The first 5 means that that the boundaries of the dark area, i.e. the boundaries of slot **83** in fig. **8**, have to be determined, and the other 5 means that both X coordinates of these boundaries must be given in the reply message.

The structure of the above described data field and the meaning of the values of the data positions of the message are unambiguously known to the smart camera, which means that it operates correctly and is able to carry out the right tasks using the right values.

After having sent the task-giving message the programmable logic asks at regular intervals whether the task has been carried out. When the smart camera has carried out the task, it edits a reply and sends it to the programmable logic. The camera's reply comprises as many data positions as requested in the queries.

Fig. **7** shows the contents of the reply. The task results of the areas are given in succession using as many data positions as needed. The reply contains answers to every question of the query in an unchanged order. In

this way it is guaranteed that the programmable logic is able to recognize the answers.

The first result comprises one data position indicating the average shade of gray in area 1. The second result gives the area of the hole darker than the average shade of gray in area **2**. This needs only one data position. Next comes the task result of area **3**. This needs two data positions as the result is the **X** and **Y** coordinate of the center. Finally we have the task result of area **4** giving the boundaries of the dark slot in area **4**, and more precisely, only the **X** coordinates. Two data positions are needed.

After the programmable logic has processed the reply, it can send a new query based on the given information. The contents of the query is naturally programmed in advance in the logic, and the program provides the message with the needed data values.

The system according to the invention can also be applied in a way that the same picture includes both the picture of the target to be examined and the reference picture. We could take as an example a continuous oven. When bread that has come out of the oven is moving to the conveyor, there is a rack beside the conveyor supporting ideally baked bread. The camera shows an area where both the ideally baked bread and bread coming from the oven are to be seen. The values of each bread on the conveyor are compared to those of the ideal bread. The aim is then to keep the breads equally dark by regulating the temperature of the oven. Because the reference bread is exposed to the same conditions as the breads to be quality-controlled, the external conditions like dirt, changes in the lighting efficiency etc. do not influence the results.

The monitor always shows the last picture, and the areas to be examined have been framed. For the user it is easy to change the place of the areas to be examined if needed, and then feed the coordinates of the areas to the programmable logic. There is no need to modify the camera software.

ANOTHER EMBODIMENT OF THE INVENTION

Fig. **11** shows another embodiment of the invention. It is used to measure the length of rod-shaped objects. The object is e.g. a metal rod coming from a cutter. It has a certain tolerance. The reference numerals are, where applicable, the same as in fig. **7**.

Rods **113** are cut in a continuous process. The rod cut-into-size is taken to a trough conveyor limited by its edges **111** and **112**. On the edge of the trough there are photocells cells **1** to **4** at a certain distance from each other. They are connected to the input of the programmable logic. At a certain distance and in the direction of the motion there is smart camera **71** having adaptation program **710** based on the invention. A servomotor can move the camera longitudinally along the trough. First, the photocell is chosen whose signal at the rear part of the rod will trigger a function according to the invention. In the figure it is cell **2**. Then the servomotor moves the camera to a point along the trough so that the front part of the rod cut-into-size lies within the vision field of the camera. An absolute sensor tells the exact location of the camera, longitudinally along the trough, to the logic. After that the process can be started.

At the moment when the rear part of rod **113** is right at photocell number **2** camera **71** takes a picture from the front part of the rod. This picture corresponds approximately to the area limited by the broken line in fig. **12**.

Now the programmable logic sends a task via the bus to the adaptation program of the smart camera. The first task is to examine, within an area $\Delta\mathbf{Y}$ in the cross direction to the trough, the X coordinates of the area having gray level higher than the average gray level of the area within ΔY. It is presupposed that the rod is darker in color than the trough. So the longitudinal position of the rod in the trough at the moment when the picture was taken can be discovered. The coordinates are sent to the programmable logic that sends the next task to the camera. The rectangular area $\Delta\mathbf{X}$ has to be examined in the direction of the coordinate Y. The coordinate of rod end **Y1** can be calculated on the basis of the gray level values of the area. The exact location of the rear end of the rod in the Y-direction is known, so that the logic program calculates the length of the rod from the above-mentioned values. If the deviation exceeds the tolerance, the rod is rejected.

If the length of the rods to be cut is changed, the process operator can easily modify the logic program and place the camera to a new position along the trough. There is no need to modify the camera software.

The above mentioned two applications serve as examples in describing the features of the invention. Naturally there are a huge number of different applications.

The Modbus protocol has been used here as an example of data transfer systems and as a process control unit the programmable logic was utilized. Of course, any other field bus protocol can be used. Instead of a bus any other data transfer connection with its protocol may be applied, e.g. connections like the Internet, the Ethernet, a radio communication, an ATM protocol etc. The adaptation program that is linked to the camera must only be made in such a way that it understands the protocol used and is able to work with it. This all is within the knowledge of a man skilled in the art. The point of this invention is the fact that once the adaptation program has been installed to the camera, the camera doesn't need any further programming. All necessary programming is made in the programmable logic by the process maintainer. Unlike in the conventional machine vision solutions there is no need for a camera programmer. The programming device may however be any programmable device, e.g. a PC.

An artisan of the art naturally understands that the programmable logic can be replaced also by a factory system, of the suppliers of which the Finnish process control system Damatic, the manufacturer Valmet Ltd, and Alcont, the manufacturer Honeywell, can be mentioned. In addition, it has to to point out that in the previous examples only gray scale vales were processed. It is clear that when a color camera is used, data calculated from different color values may be asked in the tasks. Then the reply gives information about three colors.

What is claimed is:

1. A process control system comprising:

at least one process control unit including a program for controlling operation of a process,

a video camera connected to an image-processing software producing desired image-processing results from a picture taken by the camera,

a data transfer connection along which the process control unit sends query messages for receiving the image-processing results and along which the image-processing results are transmitted to the control unit,

c h a r a c t e r i z e d in that

a query message includes an unambiguous code of the desired image-processing task and parameter values needed for performing the task,

and between the image-processing software and the data transfer connection has been placed a adaptation program that:

extracts the code and the parameter values needed for the task performance from the query message received from the data transfer connection,

transforms them to a form understood by the image-processing software so that the image-processing software is able to carry out the given image-processing task,

receives the results of the task performed by the image-processing software and

locates them in a reply message.

2. A system as in claim 1, c h a r a c t e r i z e d in that one query message may include several codes of the image-processing tasks with their parameter values.

3. A system as in claim 1, c h a r a c t e r i z e d in that the adaptation program contains several codes of the image-processing tasks, wherein in response to the codes and the attached parameter values the image-processing program is able to carry out the corresponding number of image-processing tasks

4. A system as in claim 1, c h a r a c t e r i z e d in that the image-processing software and the video camera are integrated to form a smart camera and the adaptation program has been installed in this camera.

5. A system as in claim 1, characterized in that the image-processing software and the adaptation program are installed in a computer connected to the camera.

6. A system as in claim 1, characterized in that when the process control program needs information about a picture, a query message is formed into which the code identifying the task and the related parametric values are placed.

7. A system as in claim 1, characterized in that by changing information to be received from a picture desired modifications are made only in the program for controlling operation of the process.

8. A system as in claim 1 or 7, characterized in that in any commands concerning image-processing may be included in the program for controlling operation of the process, provided that the adaptation program includes the codes identifying the tasks.

9. A system as in claim 1, characterized in that the process control unit is a programmable logic controller.

10. A system as in claim 1, characterized in that the data transfer connection is a field bus.

11. A smart camera designated for connecting via a data transfer connection to a process control system, comprising an image-processing software for processing pictures taken by the smart camera and for retrieving desired information from the picture, characterized in that:

a adaptation program containing a number of codes of image-processing tasks is arranged between a data transfer interface and the image-processing software, the codes being related to the image-processing software in such a manner that each code with its potential parameter values corresponds to an image-processing task performed by the image-processing software, and

in response to a query message from the data transfer connection the adaptation program extracts from the query message the code of an image-processing task and the parameters needed for performing the task and commands the image-processing software to carry out the specific image-processing task.

12. A smart camera as in claim 11, characterized in that the adaptation program receives the task results from the image-processing

software and locates the results into a reply message in accordance with the data transfer protocol.

13. A method for controlling video image-processing connected to a process control system comprising

at least one control unit with a process control program,

a data transfer connection via which the control unit sends query messages and receives reply messages,

a video camera with an image-processing software for obtaining desired image-processing results from a picture taken by the camera,

c h a r a c t e r i z e d by the steps of:

assigning an individual code to a desired image-processing task,

determining parameters related to the code,

arranging between the image-processing software and the data transfer connection a adaptation program which transforms the codes and the parameters to a form understood by the image-processing software in a way that the image-processing software is able to carry out the image-processing tasks linked to the codes and the parametric values related to the codes, wherein in order to carry out an image-processing task:

sending from the control unit a query message to the adaptation program, the query message containing the code of the image-processing task and the parametric values needed for task performance,

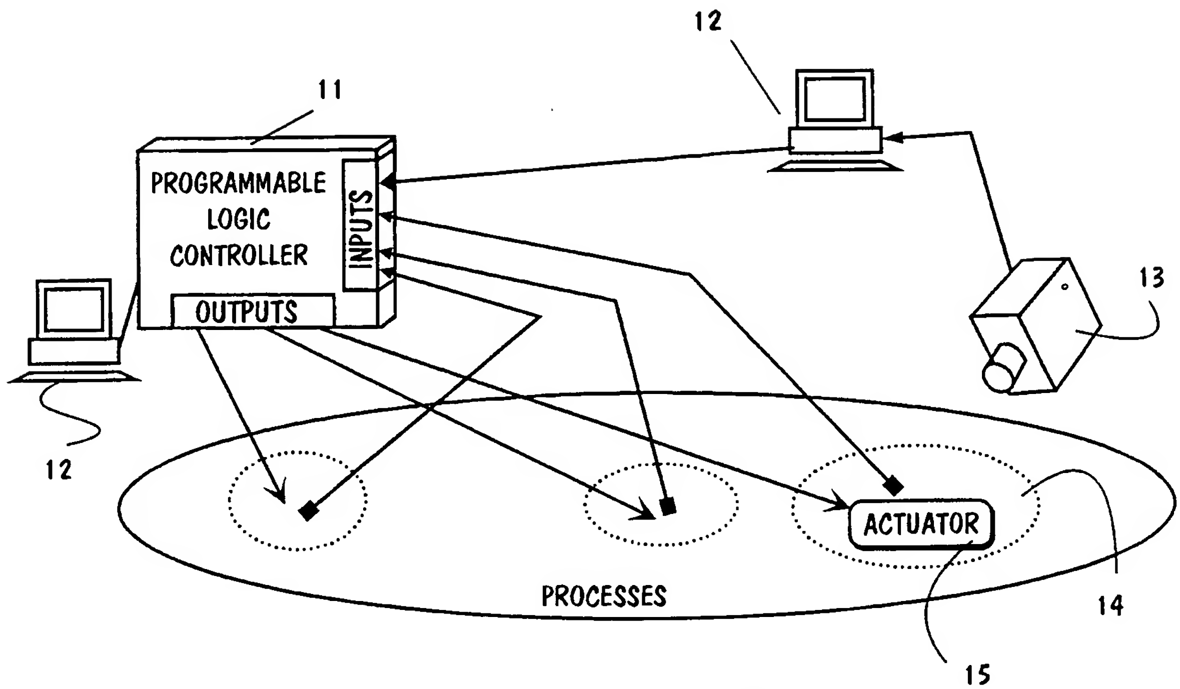
instructing by the adaptation program, in response to the query message, the image-processing program to run the task defined by the code and the parametric values,

placing the task results performed by the image-processing software into a reply message, and

sending the reply message to the control unit.

14. A method as in claim 13, c h a r a c t e r i z e d in that by changing the tasks to be performed by the image-processing program, modifications are made only in the process control program.

15. A method as in claim 13, c h a r a c t e r i z e d in that the picture taken by the video camera is displayed on a monitor and modifications needed for the process control program are made on the basis of the monitor picture.



PRIOR ART

FIG. 1

START	ADDRESS	FUNCTION	DATA	ERROR CORRECTION	END

PRIOR ART

FIG. 2

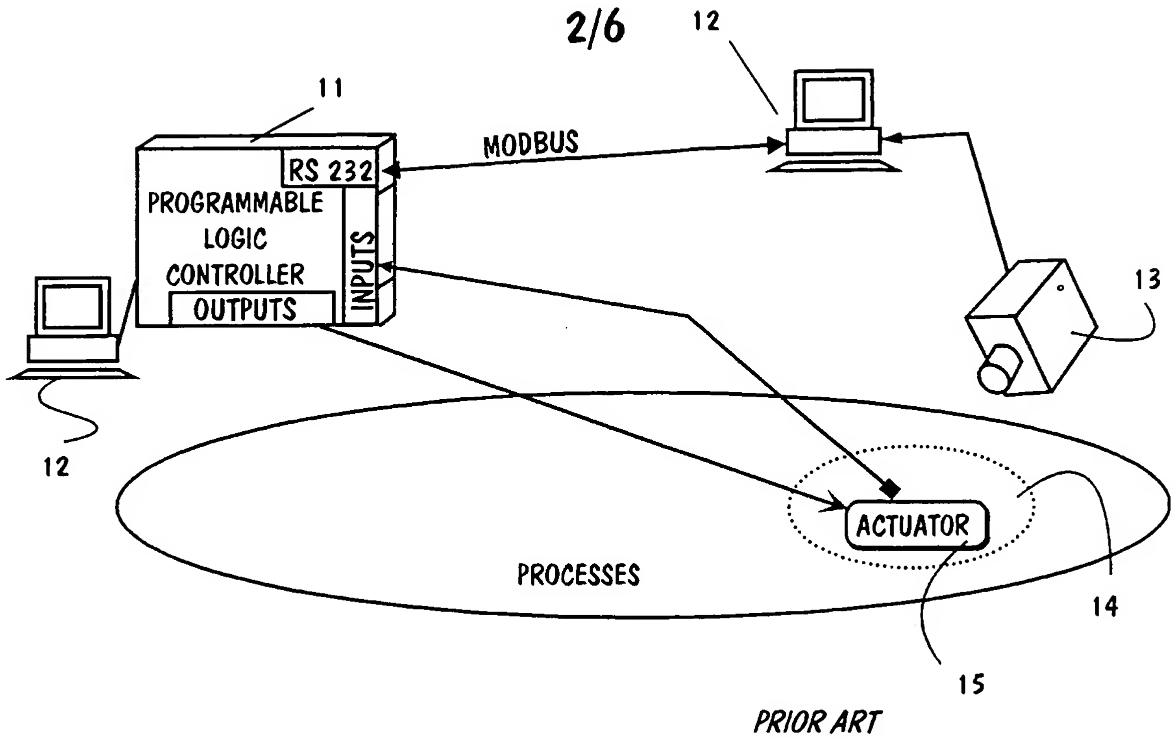

11
12
MODBUS
RS 232
PROGRAMMABLE LOGIC CONTROLLER
INPUTS
OUTPUTS
13
12
ACTUATOR
PROCESSES
14
15
PRIOR ART

Fig. 3

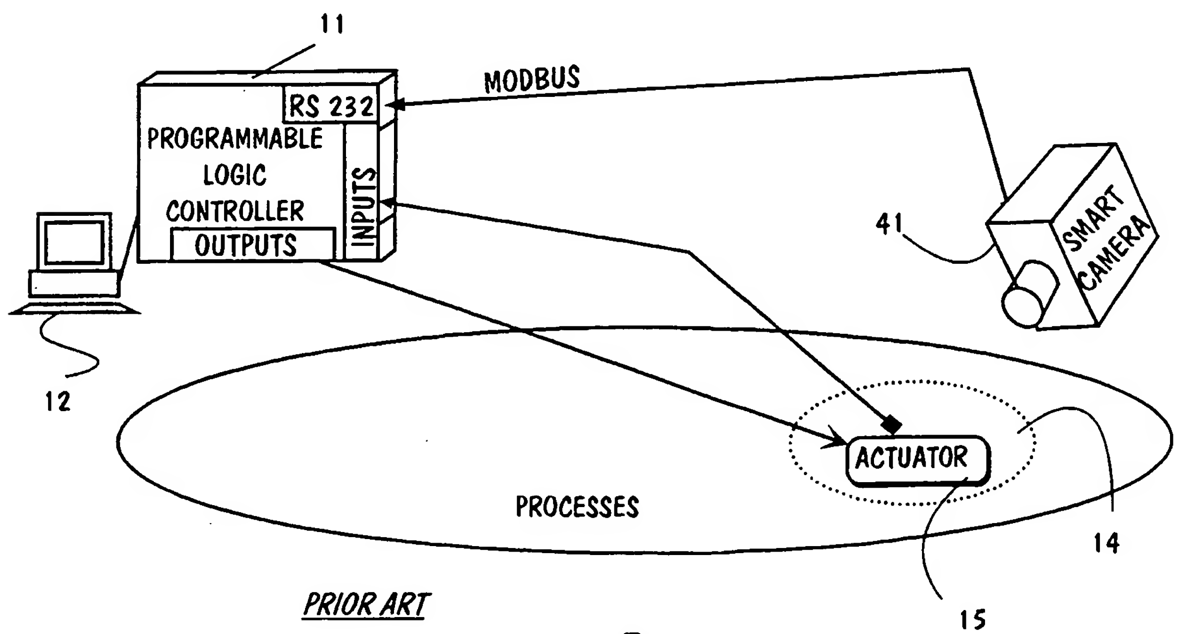

11
MODBUS
RS 232
PROGRAMMABLE LOGIC CONTROLLER
INPUTS
OUTPUTS
41
SMART CAMERA
12
ACTUATOR
PROCESSES
14
15
PRIOR ART

Fig. 4

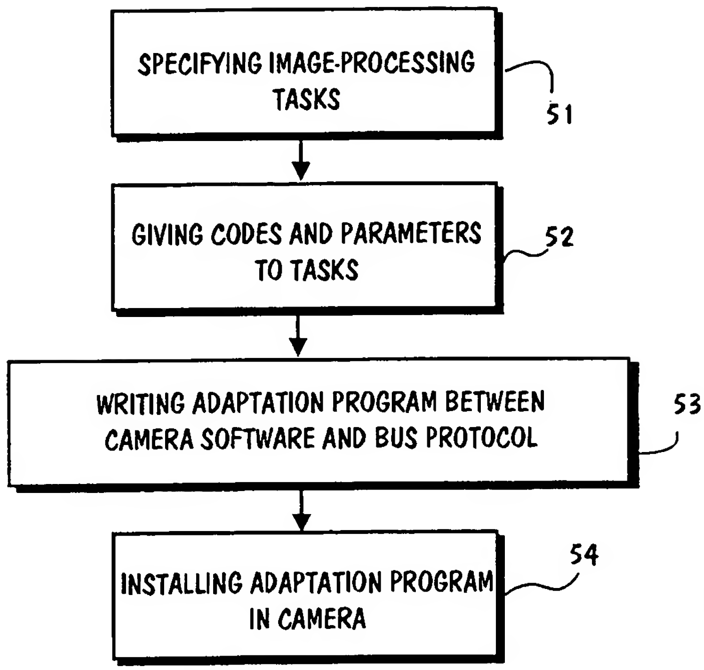

SPECIFYING IMAGE-PROCESSING TASKS
51
GIVING CODES AND PARAMETERS TO TASKS
52
WRITING ADAPTATION PROGRAM BETWEEN CAMERA SOFTWARE AND BUS PROTOCOL
53
INSTALLING ADAPTATION PROGRAM IN CAMERA
54

Fig. 5

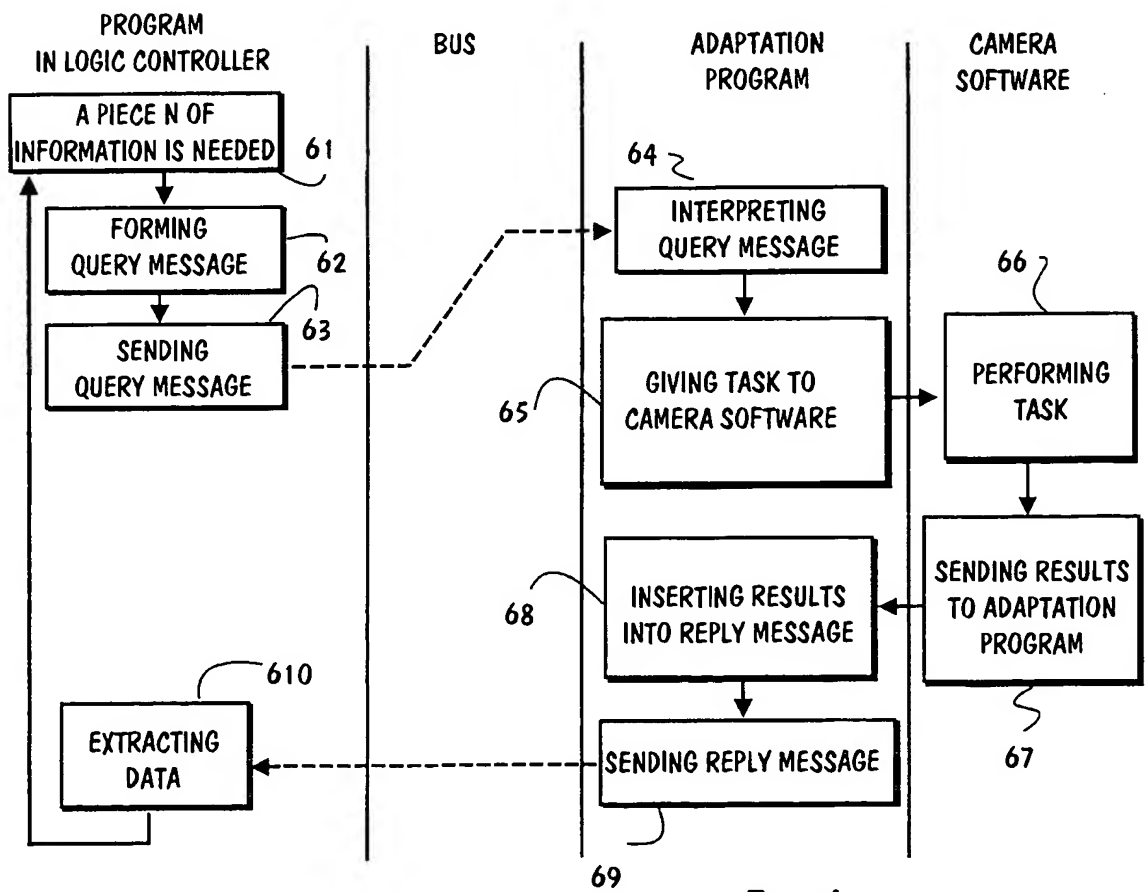

PROGRAM IN LOGIC CONTROLLER
BUS
ADAPTATION PROGRAM
CAMERA SOFTWARE
A PIECE N OF INFORMATION IS NEEDED
61
FORMING QUERY MESSAGE
62
SENDING QUERY MESSAGE
63
64
INTERPRETING QUERY MESSAGE
65
GIVING TASK TO CAMERA SOFTWARE
66
PERFORMING TASK
SENDING RESULTS TO ADAPTATION PROGRAM
67
68
INSERTING RESULTS INTO REPLY MESSAGE
610
EXTRACTING DATA
SENDING REPLY MESSAGE
69

Fig. 6

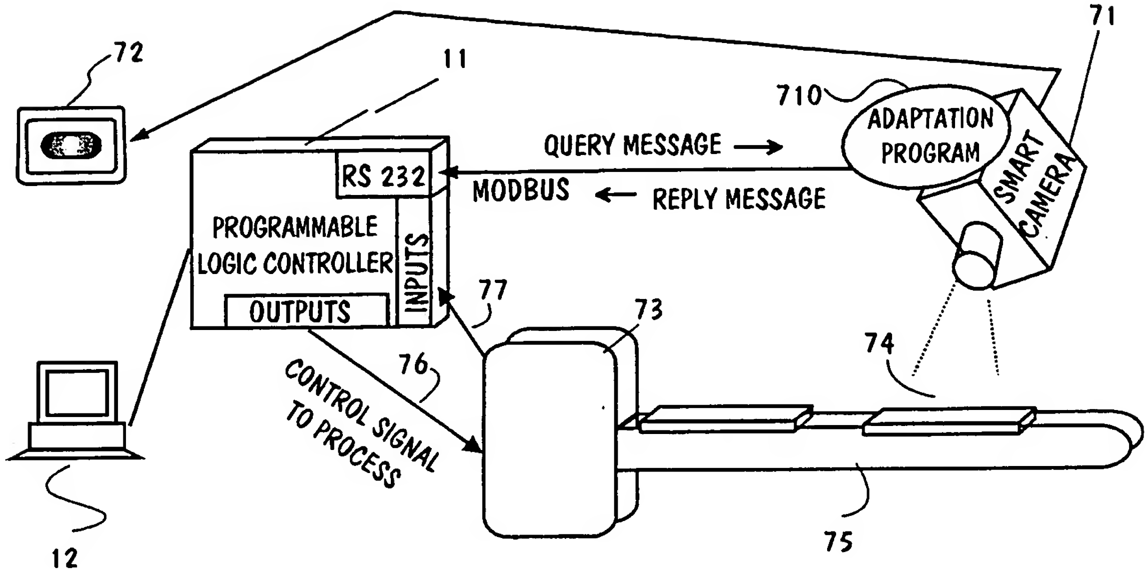

72
11
710
71
ADAPTATION PROGRAM
SMART CAMERA
QUERY MESSAGE →
RS 232
MODBUS
← REPLY MESSAGE
PROGRAMMABLE LOGIC CONTROLLER
INPUTS
OUTPUTS
77
73
74
76
CONTROL SIGNAL TO PROCESS
12
75

Fig. 7

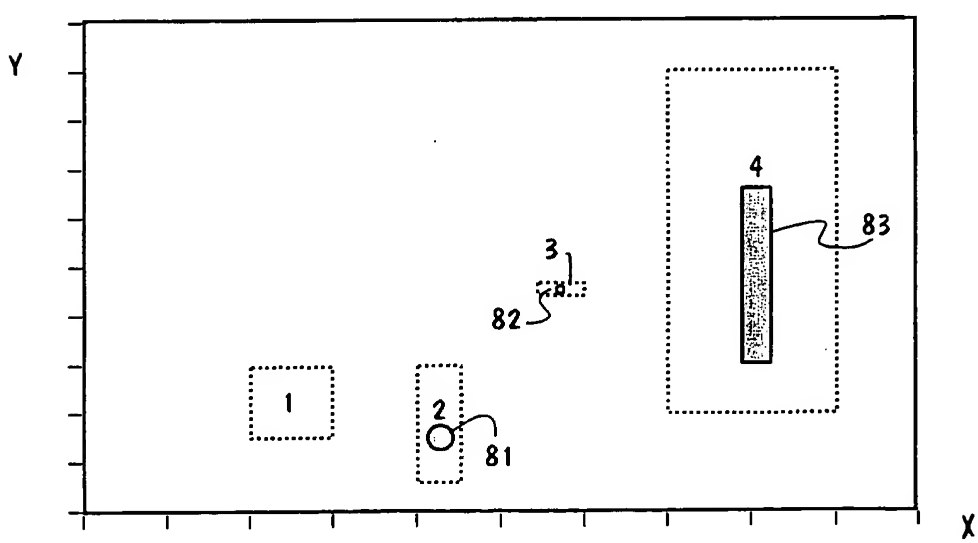

Y
X
1
2
81
3
82
4
83

Fig. 8

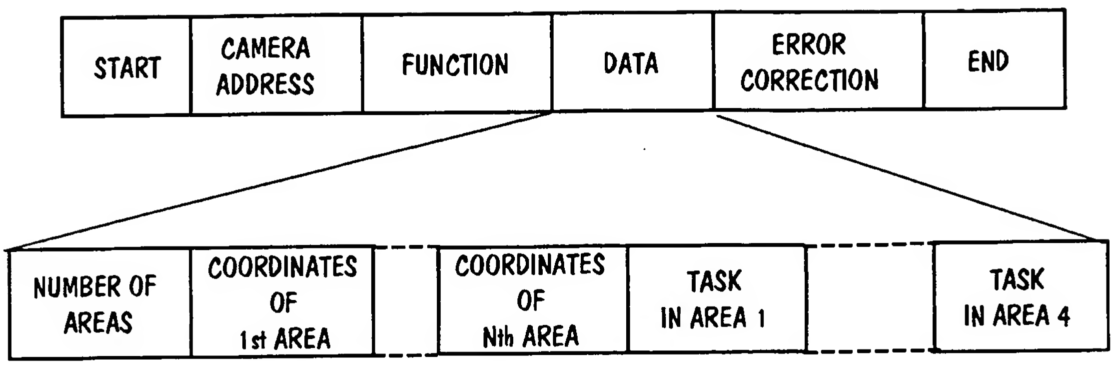

START
CAMERA ADDRESS
FUNCTION
DATA
ERROR CORRECTION
END
NUMBER OF AREAS
COORDINATES OF 1st AREA
COORDINATES OF Nth AREA
TASK IN AREA 1
TASK IN AREA 4

Fig. 9

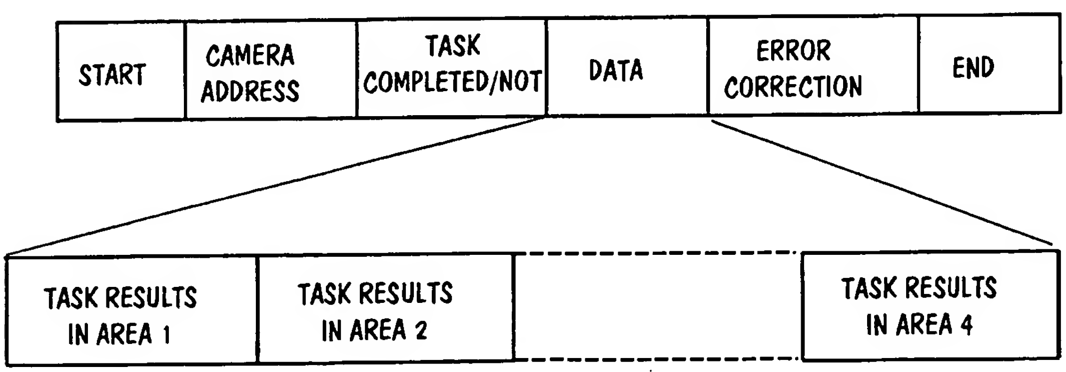
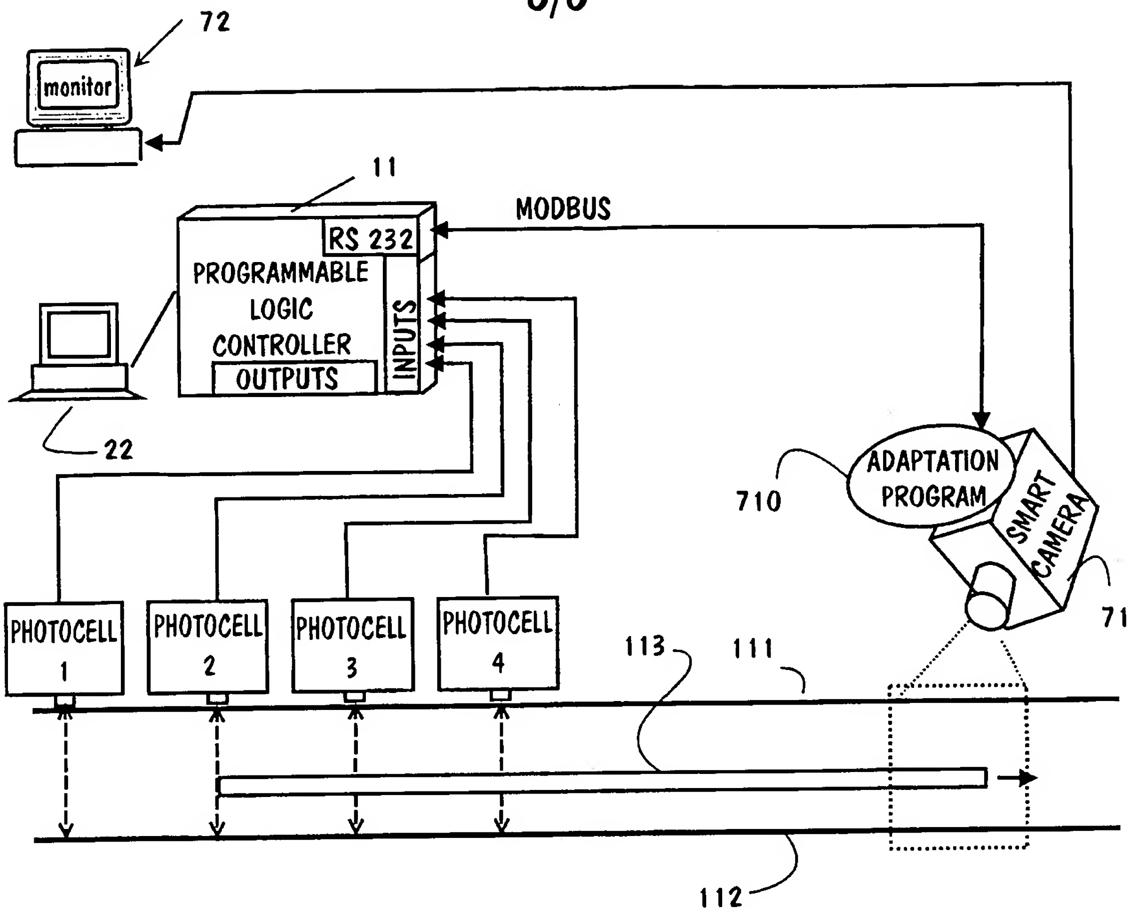
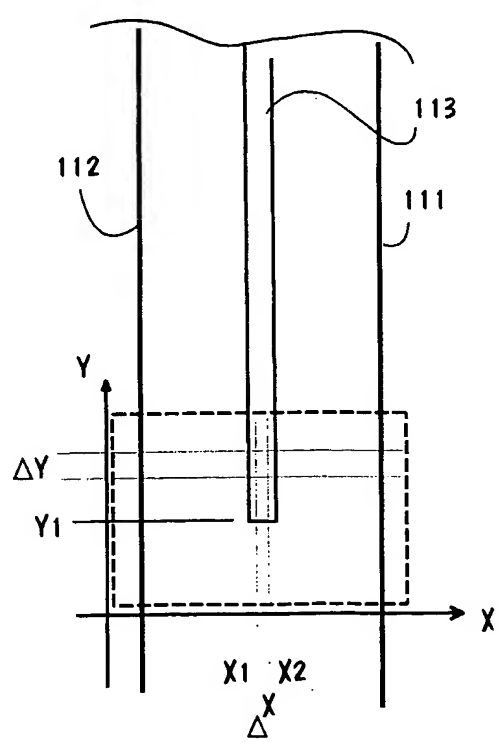

START
CAMERA ADDRESS
TASK COMPLETED/NOT
DATA
ERROR CORRECTION
END
TASK RESULTS IN AREA 1
TASK RESULTS IN AREA 2
TASK RESULTS IN AREA 4

Fig. 10



72
monitor
11
MODBUS
RS 232
PROGRAMMABLE
LOGIC
CONTROLLER
OUTPUTS
INPUTS
22
ADAPTATION
PROGRAM
SMART
CAMERA
710
71
PHOTOCELL
1
PHOTOCELL
2
PHOTOCELL
3
PHOTOCELL
4
113
111
112

Fig. 11



113
112
111
Y
ΔY
Y1
X
X1
X2
ΔX

Fig. 12